Follow the Bits: A Lightweight Instrumentation Framework for Neural Networks

Abstract

Most neural network diagnostics focus on loss values, gradient norms, activation statistics, and parameter distributions. This note proposes a complementary viewpoint: track the distribution of numerical significance throughout the network. The central question becomes:

> Where are the bits?

The approach requires only simple instrumentation and may reveal training dynamics that are largely invisible to conventional norm-based measurements.

Motivation

Neural networks operate through repeated multiplication, accumulation, normalization, and nonlinear transformation. These operations continuously move information across different magnitude scales.

Two quantities may have equal mathematical importance while occupying vastly different significance levels:

1000

and

0.001.

When combined, one dominates the other despite both being present in the computation.

Traditional metrics often conceal this behavior.

A significance-based view instead tracks how information is distributed across magnitude scales during training.

Core Idea

For any tensor X, define a significance coordinate

$$s = \log_2(|X| + \epsilon).$$

Rather than examining values directly, examine the distribution of significance levels.

Examples:

Value	Significance
1/1024	-10
1	0
1024	+10

The resulting histogram shows where numerical activity resides.

Why This Is Easy

Implementation requires only a few lines of code.

For any tensor:

s = log2(abs(x) + epsilon)

From this compute:

- Mean significance
- Significance variance
- Histograms
- Quantiles
- Entropy

No optimizer changes are required.

No architecture modifications are required.

No retraining is required.

This is purely instrumentation.

Suggested Measurements

Weight Significance

Track

$$\log_2(|W| + \epsilon)$$

for every layer.

Questions:

- How does significance evolve during training?
- Do successful models develop characteristic significance profiles?
- Does significance spread increase with depth?

Activation Significance

Track

$$\log_2(|h| + \epsilon)$$

during the forward pass.

Questions:

- Does significance drift layer by layer?
- Do residual connections stabilize significance?
- How strongly do normalization layers recenter significance?

Gradient Significance

Track

$\log_2(|g|+\epsilon)$

during backpropagation.

Questions:

- Do gradients progressively move toward lower significance levels?
- When gradients vanish, where do the bits disappear?
- When gradients explode, where does significance amplification begin?

Significance Dispersion

A simple metric is

$$D=\mathrm{Var}(\log_2(|x|+\epsilon)).$$

Interpretation:

- Small D: activity concentrated within a narrow significance band.
- Large D: activity spread across many significance levels.

Excessive dispersion may indicate that a small number of channels dominate computation while many others contribute little.

Neural Networks as Transport Systems

A useful mental model is:

- Multiplication transports significance.
- Addition merges significance.
- Normalization recenters significance.
- Quantization discards significance.
- Backpropagation transports significance backward through the network.

Under this view, training becomes a process of significance transport and redistribution.

Follow the Bits

Consider a weight update:

$w = 1000$

$\Delta w = 10^{-6}$.

The update exists mathematically.

The practical question is:

> Did any meaningful bits actually reach the weight?

Similarly, for a forward computation:

$1000 + 0.001$.

Both terms exist.

Yet most significance resides in only one of them.

The transport viewpoint asks:

> Which significance levels are actively participating in computation?

Potential Research Questions

1. Do successful and unsuccessful training runs exhibit different significance distributions?
2. Does significance dispersion increase with network depth?
3. Can significance histograms predict vanishing or exploding gradients before they become obvious in loss curves?
4. Do normalization layers function primarily as significance-centering mechanisms?
5. Can significance-aware diagnostics improve architecture design or optimizer selection?

Conclusion

The "follow the bits" perspective does not propose a new architecture, optimizer, or learning rule. It proposes a new observable.

Instead of tracking only values, track where those values reside in significance space:

$$s = \log_2(|x| + \epsilon).$$

The resulting measurements are inexpensive, easy to implement, and may provide a direct view of how numerical influence propagates through a network.

The first experiment is simple:

> Instrument a network, plot significance histograms for activations, gradients, and weights, and observe where the bits go during training.

Before proposing new mechanisms, it is worth understanding the transport processes already occurring inside existing models.